UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
07/352,530	05/15/1989	ROBERT A. FABBIO	AT989039	9777

7590	08/03/2005

WAYNE P. BAILEY
IBM CORP., INTELLECTUAL PROPERTY LAW
DEPT., 932/815, ZIP 4054
11400 BURNET ROAD
AUSTIN, TX 78758

EXAMINER
JANKUS, ALMIS R

ART UNIT	PAPER NUMBER
2672	

DATE MAILED: 08/03/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _9/24/03 decision by BPAI._
2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.
3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1_ is/are pending in the application.
 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5)☐ Claim(s) _____ is/are allowed.
6)☒ Claim(s) _1_ is/are rejected.
7)☐ Claim(s) _____ is/are objected to.
8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.
10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 a)☐ All   b)☐ Some * c)☐ None of:
  1.☐ Certified copies of the priority documents have been received.
  2.☐ Certified copies of the priority documents have been received in Application No. _____.
  3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
 * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)
2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
 Paper No(s)/Mail Date _____.
4)☐ Interview Summary (PTO-413)
 Paper No(s)/Mail Date. _____ .
5)☐ Notice of Informal Patent Application (PTO-152)
6)☐ Other: _____ .

DETAILED ACTION

1.   Claim 1 is presented:

An interface having means for presenting items for selection by a user of a data processing system, and having means for executing the selected items, said interface comprising:
      means for representing a plurality of interface objects in an object database; and
      means for dynamically associating different ones of said interface objects into a plurality of logical frame presentations based upon the data within each of said different ones of said interface objects.

2.   The indicated allowability of claim 1 is withdrawn in view of the newly

discovered reference(s) to Kolnick (US patent 5,502,839). Rejections based on the newly cited

reference(s) follow.

3.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4.   Claim 1 is rejected under 35 U.S.C. 102(e) as being anticipated by Kolnick (US

patent 5,502,839).

Claim 1 is presented in "means for" form; "means for" is modified by functional language; and "means for" is not modified by sufficient structure, material or acts for achieving the specified function.

The corresponding structure, material, or acts necessary to perform the claimed functions are identified in the written description; and equivalent prior art which performs identical functions in substantially the same way and producing substantially the same results are applied.

According to the written description, "Means for presenting items for selection by a user of a data processing system" is identified as the ASL layer which is used to present the information to the user and collect user responses. The ASL layer is a screen library layer which protects the interface shell from dealing with differences in text or graphic libraries, at Page 45 lines 16-20. Kolnick teaches the equivalent Dialog Manager which provides access to a library of "pictures" which define the menus, help texts, prompts, etc. for the Human Interface and it handles the user interaction with those pictures, at Column 20 line 65 to column 21 line 2; and further that all cultural and user idiosyncracies (such as language) are hidden from the rest of the Human Interface at column 15 lines 46-48. Also, with respect to this same limitation, the written description identifies that the system and method of this invention is an *interface* tool for displaying menus and dialogues to a user, at page 6 lines 17-19. Kolnick teaches the equivalent "In general, the Dialog Manager can be requested to load (from a file) or dynamically create (from a given specification) a picture which represents a menu, error message, help (informational) text, prompt, a set of icons, etc. This picture is usually displayed until the user responds" at column 21 lines 17-21.

Regarding the next limitation "means for executing the selected items" is identified as all work is executed out of a dialogue, at page 25 lines 28-29, and a dialogue is the actual interaction with a user, and it typically consists of attributes of the system resource. The user then enters or selects values for the attributes, completes all the responses for the dialogue, and then indicates that the user is ready to run the task which is then executed, at page 26 lines 3-9. Kolnick teaches the equivalent "The primary mechanism for interacting with the Human Interface is to move the cursor to the desired object and "select" it by pressing a key or button. An action may be performed automatically upon selection.", at column 12 lines 55-58, and "Selecting an icon usually activates the corresponding application immediately", at column 12

lines 59-60.

"Means for representing a plurality of interface objects in an object database" is identified as "menus, dialogues, and each instance of a system resource are represented as objects within an object database, and are referred to as interface objects. Data within these interface objects reflect the topology of the system resources. The interface tool traverses these interface objects based upon the data within the interface objects themselves and the user selections", at page 7 lines 1-8; and "The term, "menu" is not meant to connote any particular screen appearance, or even to suggest that items are presented in a list of any form. "Menu" will be used to signify a means of diminishing the scope of either the object or the action (the noun or the verb) which is managed. The user is in a menu as long as the user is further refining the management job. The menu could take the form of a choice from a list, objects that may be opened for further detail, zooming in our out, movement through rooms, or whatever metaphor is appropriate", at page 11 lines 9-20; and "new objects appear in the desired environment the next time that the environment is entered because they will share a common id key with the other objects in that environment and therefore be found by the interface tool", at page 9 lines 2-6; and "The id of the top menu is used as priming input to the menu processing module which uses the id to search for menu interface objects in the object data manager", at page 27 lines 1-4; further, "The field id is the id or "name" of the object. All objects (fields) that are to appear together in one dialogue must have the same id", at page 19 lines 18-20; and "All objects found to meet the id key are sent for display to a screen library", at page 27 lines 6-8. The equivalents taught at Kolnick are as follows: "Interface objects", as used in the instant application are equivalent to "processes" in Kolnick. "A "process" is a self-contained package of data and executable procedures which operate on that data", at column 5 lines 19-22; "Any process can create context processes. Each new context thus defined is completely contained inside the context in which it was created" at column 5 line 66 to column 6 line 1, "This "nesting" allows the name space to be structured hierarchically to any desired depth", at column 6 lines 2-3. "Menu", as used in the instant application is equivalent to "metaphor" and "context" in Kolnick. "A "metaphor" defines the visual environment in which the user operates on a particular screen. It consists of any combination of pre-defined windows, icons, menus, and soft-keys appropriate to that environment. In general, a metaphor graphically depicts a real

user environment. Thus the icons may represent physical objects in the user's frame of reference, such as file folders or diskettes, menus and messages phrased in familiar terminology, and so on", at column 39 lines 23-30; further, "Every process also has a non-unique, symbolic "name".", column 5 line 42; "the system-wide name space is partitioned into distinct subsets by means of "contexts" ", at column 7 lines 30-31; "A context is simply a collection of related processes", at column 7 lines 32-33. Also, "The context nesting level determines the "scope of reference" when sending messages between processes by name", at column 6 lines 37-38; "the contexts are searched from the current context upward until a match is found. All processes with the given name at that level are then sent a copy of the message", at column 9 lines 21-25. "A "metaphor" picture comprises more-or-less arbitrary picture elements which model a particular frame of reference for the user. For example, the picture may represent a "desktop", with appropriate elements (typewriter, letter "in" and "out" trays, pads of paper, etc.). The name of the metaphor must be unique among metaphors", column 41 lines 14-19; and "Menus (as well as icons, prompts, and information) can be stored in "libraries" to which the metaphor may be linked when it is built or when it is initiated. A library consists of individual elements, each of which represents one menu, icon, etc. The first substring of the element's "tag" field is the element's name. The "name" is referenced in the corresponding dialog request ("icon", etc.) or response ("click"). ", at column 42 lines 38-44.

The last limitation of "means for dynamically associating different ones of said interface objects into a plurality of logical frame presentations based upon the data within each of said different ones of said interface objects" is identified as the data within each of interface objects is the representation of the hierarchical relationship within the interface objects. This is deduced from the recitations of claims 3 and 4 which are part of the instant specification. Kolnick teaches that the hierarchy consists of processes and contexts. The hierarchy of the system-wide name space and inter-process communication is taught at figures 4-5 and at column 7 line 5 to column 10 line 33. "Processes are referenced without regard to their physical location via a small set of message-passing primitives. Every process has both a unique system-generated identifier and a not necessarily unique name. The identifier provides quick direct access, while the name has a limited scope and provides symbolic, indirect access.

Column 6 lines 52-58", and that the dynamic association among processes and contexts includes, among others, the use of messages (data which tells a process what to do and/or supplies it with information it needs to carry out its operation); unique process identifiers (PID); non-unique symbolic names; pointers for direct access; and user input.

The "logical frame presentation" is identified as "The collection of objects sent to the screen library from the interface tool is referred to herein as the *logical frame presentation*, since it has not yet been displayed to the user.", at page 27 lines 10-13. The equivalent taught at Kolnick is "Icons are small pictures which represent applications or services and are organized into sets (or "frames of reference") of related functions", at column 22 lines 22-24; "ICON displays a new icon in the current metaphor, and it may contain information specifying the name of a picture element in the metaphor's current icon library; the identity of the icon on the screen; and a single picture element", at column 39 lines 63-67; "There is one Draw Manager per console, and it provides access to a library of "pictures" which define the menus, help, prompts, etc., for the Human Interface (and possibly the rest of the system), and it handles the user interaction with those pictures", at column 22 lines 33-37; "Menus (as well as icons, prompts, and information) can be stored in "libraries" to which the metaphor may be linked when it is built or when it is initiated", at column 42 lines 38-40"; and "Library references can be built into a metaphor picture", at column 42 line 48".

Kolnick teaches additional ways of dynamically associating different ones of said interface objects into a plurality of logical frame presentations based upon the data within each of said different ones of said interface objects, at Column 5 lines 19-29, Column 8 lines 13-21, Column 11 lines 8-15, Column 12 lines 44-67, Column 14 line 65 to column 15 line 7, Column 18 line 62 to column 19 line 2, Column 20 lines 23-30, Column 21 lines 17-21, Column 22 lines 15-21, Column 26 lines 12-19 and lines 40-43, Column 29 line 65 to column 30 line 10, Column 36 lines 63-67, Column 41 lines 41-49 and lines 53-64, and Column 43 line 66 to column 44 line 5.

Below, is a table to facilitate easier comprehension of claimed features, corresponding material from the written description, and Kolnick prior art equivalents.

The claimed feature is at the top of the table in bold; excerpts from the written description are in the left column; Kolnick equivalent teachings are in the right column.

Means for presenting items for selection by a user of a data processing system

the ASL layer is used to *present the information to the user and collect user responses*. The ASL layer 'is a *screen library layer* which *protects the interface shell from dealing with differences in text or graphic libraries*. Page 45 lines 16-20	*Dialog Manager*...provides access to a *library of "pictures" which define the menus, help texts, prompts, etc.* for the Human Interface and it handles the user interaction with those pictures. Column 20 line 65 to column 21 line 2 All cultural and user idiosyncracies (such as language) are hidden from the rest of the Human Interface. Column 15 lines 46-48
The system and method of this invention is an *interface* tool for displaying menus and dialogues to a user Page 6 lines 17-19	In general, the Dialog Manager can be requested to load (from a file) or dynamically create (from a given specification) a picture which represents a menu, error message, help (informational) text, prompt, a set of icons, etc. This picture is usually displayed until the user responds. Column 21 lines 17-21

means for executing the selected items

| All work is executed out of a dialogue. Page 25 lines 28-29 | The primary mechanism for interacting with the Human Interface is to move the cursor to the desired object and "select" it by pressing a key or button. An action may be performed automatically upon selection. Column 12 lines 55-58 |
| A dialogue is the actual interaction with a user, and it typically consists of attributes of the system resource. The user then enters or selects values for the attributes, completes all the responses for the dialogue, and then indicates that the user is ready to run the task which is then executed Page 26 lines 3-9 | Selecting an icon usually activates the corresponding application immediately. Column 12 lines 59-60 |

means for representing a plurality of interface objects in an object database

menus, dialogues, and each instance of a system resource are represented as objects within an object database, and are referred to as interface objects. Data within these interface objects reflect the topology of the system resources. The interface tool traverses these interface objects based upon the data within the	"Interface objects", as used in the instant application are equivalent to "processes" in Kolnick.
	a "process" is a self-contained package of data and executable procedures which operate on that data. Column 5 lines 19-22
	Any process can create context processes. Each new context thus defined is completely contained inside the context in which it was created.

interface objects themselves and the user selections. Page 7 lines 1-8	Column 5 line 66 to column 6 line 1 This "nesting" allows the name space to be structured hierarchically to any desired depth. Column 6 lines 2-3
The term, "menu" is not meant to connote any particular screen appearance, or even to suggest that items are presented in a list of any form. "Menu" will be used to signify a means of diminishing the scope of either the object or the action (the noun or the verb) which is managed. The user is in a menu as long as the user is further refining the management job. The menu could take the form of a choice from a list, objects that may be opened for further detail, zooming in our out, movement through rooms, or whatever metaphor is appropriate. Page 11 lines 9-20	"Menu", as used in the instant application is equivalent to "metaphor" and "context" in Kolnick. A "metaphor" defines the visual environment in which the user operates on a particular screen. It consists of any combination of pre-defined windows, icons, menus, and soft-keys appropriate to that environment. In general, a metaphor graphically depicts a real user environment. Thus the icons may represent physical objects in the user's frame of reference, such as file folders or diskettes, menus and messages phrased in familiar terminology, and so on. Column 39 lines 23-30
new objects appear in the desired environment the next	Every process also has a non-unique, symbolic "name".

time that the environment is entered because they will share a common id key with the other objects in that environment and therefore be found by the interface tool. Page 9 lines 2-6	Column 5 line 42 the system-wide name space is partitioned into distinct subsets by means of "contexts" column 7 lines 30-31 A context is simply a collection of related processes Column 7 lines 32-33
The id of the top menu is used as priming input to the menu processing module which uses the id to search for menu interface objects in the object data manager Page 27 lines 1-4	The context nesting level determines the "scope of reference" when sending messages between processes by name. Column 6 lines 37-38 The contexts are searched from the current context upward until a match is found. All processes with the given name at that level are then sent a copy of the message Column 9 lines 21-25
The field id is the id or "name" of the object. All objects (fields) that are to appear together in one dialogue must have the same id Page 19 lines 18-20	A "metaphor" picture comprises more-or-less arbitrary picture elements which model a particular frame of reference for the user. For example, the picture may represent a "desktop", with appropriate elements (typewriter, letter "in" and "out" trays, pads of paper, etc.). The name of the metaphor must be unique among metaphors. Column 41 lines 14-19

All objects found to meet the id key are sent for display to a screen library Page 27 lines 6-8	Menus (as well as icons, prompts, and information) can be stored in "libraries" to which the metaphor may be linked when it is built or when it is initiated. A library consists of individual elements, each of which represents one menu, icon, etc. The first substring of the element's "tag" field is the element's name. The "name" is referenced in the corresponding dialog request ("icon", etc.) or response ("click"). Column 42 lines 38-44

means for dynamically associating different ones of said interface objects into a plurality of logical frame presentations based upon the data within each of said different ones of said interface objects

interface objects are dynamically associated based upon the data within each of the interface objects. Limitation from claim 1	The hierarchy consists of processes and contexts. The hierarchy of the system-wide name space and inter-process communication is taught at figures 4-5 and at column 7 line 5 to column 10 line 33.
The data within each of interface objects is the	Processes are referenced without regard to their physical location via a small set of message-passing primitives. Every process has both a unique system-

representation of the hierarchical relationship within the interface objects. Deduced from claims 3 and 4.	generated identifier and a not necessarily unique name. The identifier provides quick direct access, while the name has a limited scope and provides symbolic, indirect access. Column 6 lines 52-58 The dynamic association among processes and contexts includes, among others, the use of messages (data which tells a process what to do and/or supplies it with information it needs to carry out its operation); unique process identifiers (PID); non-unique symbolic names; pointers for direct access; and user input. Column 5 lines 19-29 Column 8 lines 13-21 Column 11 lines 8-15 Column 12 lines 44-67 Column 14 line 65 to column 15 line 7 Column 18 line 62 to column 19 line 2 Column 20 lines 23-30 Column 21 lines 17-21 Column 22 lines 15-21 Column 26 lines 12-19 and lines 40-43 Column 29 line 65 to column 30 line 10 Column 36 lines 63-67 Column 41 lines 41-49 and lines 53-64 Column 43 line 66 to column 44 line 5
The collection of objects sent to the screen library from the	Icons are small pictures which represent applications or services and are organized into sets (or "frames of

interface tool is referred to herein as the *logical frame presentation*, since it has not yet been displayed to the user. Page 27 lines 10-13	reference") of related functions. Column 22 lines 22-24 ICON displays a new icon in the current metaphor, and it may contain information specifying the name of a picture element in the metaphor's current icon library; the identity of the icon on the screen; and a single picture element. Column 39 lines 63-67 There is one Draw Manager per console, and it provides access to a library of "pictures" which define the menus, help, prompts, etc., for the Human Interface (and possibly the rest of the system), and it handles the user interaction with those pictures Column 22 lines 33-37 Menus (as well as icons, prompts, and information) can be stored in "libraries" to which the metaphor may be linked when it is built or when it is initiated. Column 42 lines 38-40 Library references can be built into a metaphor picture Column 42 line 48

5. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Almis R Jankus whose telephone number is 571-272-7643. The examiner can normally be reached on M-F, 8-5.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Michael Razavi can be reached on 571-272-7664. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

AJ

MICHAEL RAZAVI
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2600